UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/720,585	11/24/2003	Erik R. Altman	YOR920030405US1	5059

29683 7590 03/15/2006

HARRINGTON & SMITH, LLP
4 RESEARCH DRIVE
SHELTON, CT 06484-6212

EXAMINER	
LAI, VINCENT	

ART UNIT	PAPER NUMBER
2181	

DATE MAILED: 03/15/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/720,585	ALTMAN ET AL.	
	Examiner	Art Unit	
	Vincent Lai	2181	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *24 November 2003*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-44* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-44* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *11/24/2003* is/are: a)☐ accepted or b)☒ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

FRITZ FLEMING
Supervisory PRIMARY EXAMINER
GROUP 2100

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *3/29/2004*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Information Disclosure Statement

1. The information disclosure statement (IDS) submitted on March 29, 2004 was considered by the examiner.

Drawings

2. The drawings are objected to as failing to comply with 37 CFR 1.84(p)(5) because they include the following reference character(s) not mentioned in the description: Element 22A of figure 4. It is acknowledged that reference to element 22A is made in the specification but it is not explicitly stated. Corrected drawing sheets in compliance with 37 CFR 1.121(d), or amendment to the specification to add the reference character(s) in the description in compliance with 37 CFR 1.121(b) are required in reply to the Office action to avoid abandonment of the application. Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either "Replacement Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d). If the changes are not accepted by the examiner, the applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Claim Objections

3. Claims 4 and 25 are objected to because of the following informalities: It is suggested the claim be changed to "...at least one page table entry form **a** translation lookaside buffer (TLB)". Appropriate correction is required.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

4. Claims 2, 7-8, 20-21, and 41-42 are rejected for the following reasons:

Claim 2 recites the limitation "said program instructions" in line 3 of the claim. There is insufficient antecedent basis for this limitation in the claim. It appears "said program instructions" either has a specific or a unique use, which was not claimed, thus changing the intended limitations and rendering the claim vague and indefinite.

Claims 7-8 are rejected because of their dependency on claim 2.

Claim 20 recites the limitation "said at least one extension" in line 2 of the claim. There is insufficient antecedent basis for this limitation in the claim. It appears "said at least one extension" is a limitation that has been omitted, and thus renders the claim vague and indefinite.

Claim 21 is rejected because of its dependency on claim 20.

Claim 41 recites the limitation "said at least one extension" in line 2 of the claim. There is insufficient antecedent basis for this limitation in the claim. . It appears "said at least one extension" is a limitation that has been omitted, and thus renders the claim vague and indefinite.

Claim 42 is rejected because of its dependency on claim 41.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claims 1-44 are rejected under 35 U.S.C. 102(b) as being anticipated by Yates, Jr. et al (U.S. Patent # 6,397,379 B1), herein referred to as Yates, Jr. et al.

As per claim 1, Yates, Jr. et al discloses a digital data processor (See column 3, lines 14-15: Same as a microprocessor) comprising an instruction unit (See column 3, lines 15-20), said instruction unit comprising a code page (See

column 73, lines 28-30: Discloses the existence of code pages) that is partitioned for storing in a first section thereof a plurality of instruction words and, in association with at least one instruction word, in a second section thereof an extension to said at least one instruction word (See column 2, lines 33-36 and column 3, lines 15-20: The instruction unit can operate with two sources of instructions and pages can be partitioned to handle two sets of instructions).

As per claim 2, Yates, Jr. et al discloses where said first section is comprised of a first plurality of contiguous storage locations, and where said second section is comprised of a second plurality of contiguous storage locations (See column 3, lines 15-20 and column 88, lines 25-26: The contiguous storage locations are known as sectors and from the previous claim, there are two sections), and where said program instructions are one of fixed length and variable length program instructions (See column 22, lines 21-29, and 24, lines 3-7: Variable length instructions are handled. Fixed length instructions and variable length instructions are inherent in Intel X86 processors disclosed by Yates, Jr.).

As per claim 3, Yates, Jr. et al discloses further comprising at least one page table entry bit having a state for indicating, on a code page by code page basis, whether the code page is partitioned into said first and second sections for storing instruction words and at, least one instruction word extension, or whether the code page is comprised instead of a single section storing only instruction

words (See column 2, lines 36-49: There are flags for both switching from one set of instructions to another and from one storage section to another).

As per claim 4, Yates, Jr. et al discloses where said at least one page table entry bit is output from translation lookaside buffer (TLB) (See column 30, lines 61-67 and column 94, lines 23-28: The XP bit present in a page entry is dependent on the TLB).

As per claim 5, Yates, Jr. et al discloses further comprising address fault circuitry for determining, in accordance with a state of the at least one page table entry bit, whether a generated instruction address is a valid address for the code page (See column 74, lines 35-47: Probing is done to determine validity of addresses).

As per claim 6, Yates, Jr. et al discloses further comprising address circuitry for addressing an instruction word in said first section using a current instruction address, while simultaneously addressing an extension to said instruction word at a fixed offset from said current instruction address. (See column 61, lines 15-20: Instructions and an offset can be accessed at the same time).

As per claim 7, Yates, Jr. et al discloses where at least some of the second storage locations are not allocated for storing instruction word extensions

(See column 3, lines 15-20: The second storage location can be used to store data instead of instructions).

As per claim 8, Yates, Jr. et al discloses where said at least some of the second storage locations that are not allocated for storing instruction word extensions are allocated instead for storing at least one of constant values, security information, and error detection and/or correction information for the code page (See column 3, lines 15-20: The second storage location can be used to store data instead of instructions).

As per claim 9, Yates, Jr. et al discloses further comprising an address comparator for detecting when program execution has reached the end of the first section for ensuring that a next instruction address is not contained in the second section (See column 2, lines 36-51: There are flags for switching from one storage section to another and there is available information on the data in the storage sections).

As per claim 10, Yates, Jr. et al teaches where each instruction word has a width of x bits (See column 24, lines 3-7: Instruction length is variable), where each extension has a width of y bits (See column 24, lines 3-7: Variable instruction length can be viewed as instruction length of x + variable length y) where x=n(8-bits) (See column 24, lines 3-7 and column 38, lines 61-65: Yates, Jr. et al discloses a 64-bit X86 Intel processor. It is inherent that X86 Intel

processors are backwards compatible and can handle standard bit size instructions as well as variable length instructions), where y=m(8-bits), where n is an integer greater than one, and where m has a value less than one, equal to one, or greater than one (See column 24, lines 3-7: Variable length instructions are can be viewed as regular length instructions with extensions, include ones of any length).

As per claim 11, Yates, Jr. et al discloses further comprising circuitry, coupled to an output of said code page, to combine an addressed instruction word read out of said code page with a corresponding instruction word extension that is also read out of said code page (See column 61, lines 44-50: An address is calculated with a byte offset).

As per claim 12, Yates, Jr. et al discloses where said combining circuitry comprises an instruction cache having a bit width w at least equal to a width of an instruction word plus a width of the instruction word extension (See column 87, lines 31-33: Each cache line is 256 bits, which is plenty given lengths discussed in specifications).

As per claim 13, Yates, Jr. et al discloses where said combining circuitry comprises an instruction cache having a bit width w at least equal to a width of an instruction word plus a width of the instruction word extension (See column 87, lines 31-33: Each cache line is 256 bits, which is plenty given lengths

discussed in specifications), said instruction cache having an output coupled to an input stage of an instruction pipeline (See figure 1C, the Instruction Cache 112 is part of the pipeline stage the fetches the instruction from the global bus), said input stage having a bit width of w (See column 24, lines 3-7: Instructions can be variable length).

As per claim 14, Yates, Jr. et al discloses where said combining circuitry comprises an input stage of an instruction pipeline (Fetch 110, see figure 1C).

As per claim 15, Yates, Jr. et al discloses where said combining circuitry comprises an instruction decode stage of an instruction pipeline (Instruction Decode and Dispatch unit 140, see figure 1C).

As per claim 16, Yates, Jr. et al discloses further comprising circuitry, coupled to an output of said code page, to selectively combine, in response to the state of said at least one page table entry bit, an addressed instruction word read out of said code page with a corresponding instruction word extension that is also read out of said code page (See column 61, lines 44-50: An address is calculated with a byte offset).

As per claim 17, Yates, Jr. et al discloses where said combining circuitry comprises a multiplexer (Mux 784, see figure 7H) having a first set of inputs coupled to an instruction word extension output of said code page and a second

set of inputs coupled to an invalid instruction word extension (See figure 7H: The mux 784 takes in multiple sectors and chooses which sector 722 to output).

As per claim 18, Yates, Jr. et al discloses where said combining circuitry comprises an instruction decode stage of an instruction pipeline (Instruction Decode and Dispatch unit 140, see figure 1C).

As per claim 19, Yates, Jr. et al discloses where said instructions comprise Reduced Instruction Set Computer (RISC) instructions (See column 22, 21-25: The processor disclosed is a RISC processor and thus will handle RISC instructions).

As per claim 20, Yates, Jr. et al discloses where said RISC instructions (See column 22, 21-25: The processor disclosed is a RISC processor and thus will handle RISC instructions) have a width of 32-bits (See column 38, lines 61-65: Yates, Jr. et al discloses a 64-bit X86 Intel processor. It is inherent that X86 Intel processors are backwards compatible and can handle standard bit size instructions as well as variable length instructions), where said at least one extension has a width of 8-bits (See column 24, lines 3-7: Variable length instructions are can be viewed as regular length instructions with extensions, include ones of 8-bits).

As per claim 21, Yates, Jr. et al discloses where said code page has a storage capacity of 4096 bytes (See column 61, lines 49-50), where said first section comprises 3072 bytes, and where said second section comprises 1024 bytes (See column 2, lines 36-49: Sections in pages are determined by flags and thus can be set to partition the 4096 bytes into 3072 and 1024 bytes).

As per claim 22, Yates, Jr. et al discloses a method to operate an instruction unit (See column 3, lines 15-20) having a code page, comprising:

partitioning said code page (See column 73, lines 28-30: Discloses the existence of code pages) into at least two sections (See column 2, lines 36-49: Sections in pages are determined by flags and thus can be set to partitions);

and storing in a first section thereof a plurality of instruction words and, in association with at least one instruction word, in a second section thereof an extension to said at least one instruction word (See column 2, lines 33-36 and column 3, lines 15-20: The instruction unit can operate with two sources of instructions and pages can be partitioned to handle two sets of instructions).

As per claim 23, Yates, Jr. et al discloses where said first section is comprised of a first plurality of contiguous storage locations, and where said second section is comprised of a second plurality of contiguous storage locations (See column 3, lines 15-20 and column 88, lines 25-26: The contiguous storage locations are known as sectors and from the previous claim, there are two sections), and where said program instructions are one of fixed length and

variable length program instructions (See column 24, lines 3-7: Variable length instructions are handled. Fixed length instructions are inherent).

As per claim 24, Yates, Jr. et al discloses further comprising setting a state of at least one page table entry bit for indicating, on a code page by code page basis, whether the code page is partitioned into said first and second sections for storing instruction words and at, least one instruction word extension, or whether the code page is comprised instead of a single section storing only instruction words (See column 2, lines 36-49: There are flags for both switching from one set of instructions to another and from one storage section to another).

As per claim 25, Yates, Jr. et al discloses further comprising outputting said at least one page table entry bit from translation lookaside buffer (TLB) (See column 30, lines 61-67 and column 94, lines 23-28: The XP bit present in a page entry is dependent on the TLB).

As per claim 26, Yates, Jr. et al discloses further comprising determining, in accordance with a state of the at least one page table entry bit, whether a generated instruction address is a valid address for the code page (See column 74, lines 35-47: Probing is done to determine validity of addresses).

As per claim 27, Yates, Jr. et al discloses further comprising addressing an instruction word in said first section using a current instruction address, while

simultaneously addressing an extension to said instruction word at a fixed offset from said current instruction address. (See column 61, lines 15-20: Indicative of the ability to access both an instruction and an offset at the same time)

As per claim 28, Yates, Jr. et al discloses where at least some of the second storage locations are not allocated for storing instruction word extensions (See column 3, lines 15-20: The second storage location can be used to store data instead of instructions).

As per claim 29, Yates, Jr. et al discloses where said at least some of the second storage locations that are not allocated for storing instruction word extensions are allocated instead for storing at least one of constant values, security information, and error detection and/or correction information for the code page (See column 3, lines 15-20: The second storage location can be used to store data instead of instructions).

As per claim 30, Yates, Jr. et al discloses further comprising detecting when program execution has reached the end of the first section for ensuring that a next instruction address is not contained in the second section (See column 2, lines 36-51: There are flags for switching from one storage section to another and there is available information on the data in the storage sections).

As per claim 31, Yates, Jr. et al teaches where each instruction word has a width of x bits (See column 24, lines 3-7: Instruction length is variable), where each extension has a width of y bits (See column 24, lines 3-7: Variable instruction length can be viewed as instruction length of x + variable length y) where x=n(8-bits) (See column 24, lines 3-7 and column 38, lines 61-65: Yates, Jr. et al discloses a 64-bit processor, but it is well known in the art that backwards compatibility is built into processors and thus can handle standard bit size instructions as well as variable length instructions), where y=m(8-bits), where n is an integer greater than one, and where m has a value less than one, equal to one, or greater than one (See column 24, lines 3-7: Variable length instructions are can be viewed as regular length instructions with extensions, include ones of any length).

As per claim 32, Yates, Jr. et al discloses further comprising circuitry combining an addressed instruction word read out of said code page with a corresponding instruction word extension that is also read out of said code page (See column 61, lines 44-50: An address is calculated with a byte offset).

As per claim 33, Yates, Jr. et al discloses where combining operates an instruction cache having a bit width w at least equal to a width of an instruction word plus a width of the instruction word extension (See column 87, lines 31-33: Each cache line is 256 bits, which is plenty given lengths discussed in specifications).

As per claim 34, Yates, Jr. et al discloses where combining operates an instruction cache having a bit width w at least equal to a width of an instruction word plus a width of the instruction word extension (See column 87, lines 31-33: Each cache line is 256 bits, which is plenty given lengths discussed in specifications), said instruction cache having an output coupled to an input stage of an instruction pipeline (See figure 1C, the Instruction Cache 112 is part of the pipeline stage the fetches the instruction from the global bus), said input stage having a bit width of w (See column 24, lines 3-7: Instructions can be variable length).

As per claim 35, Yates, Jr. et al discloses where combining occurs at an input stage of an instruction pipeline (Fetch 110, see figure 1C).

As per claim 36, Yates, Jr. et al discloses where combining occurs at an instruction decode stage of an instruction pipeline (Instruction Decode and Dispatch unit 140, see figure 1C).

As per claim 37, Yates, Jr. et al discloses further comprising circuitry, in response to the state of said at least one page table entry bit, an addressed instruction word read out of said code page with a corresponding instruction word extension that is also read out of said code page (See column 61, lines 44-50: An address is calculated with a byte offset).

As per claim 38, Yates, Jr. et al discloses where selectively combining comprises operating a multiplexer (Mux 784, see figure 7H) having a first set of inputs coupled to an instruction word extension output of said code page and a second set of inputs coupled to an invalid instruction word extension (See figure 7H: The mux 784 takes in multiple sectors and chooses which sector 722 to output).

As per claim 39, Yates, Jr. et al discloses where selectively combining circuitry comprises operating an instruction decode stage of an instruction pipeline (Instruction Decode and Dispatch unit 140, see figure 1C).

As per claim 40, Yates, Jr. et al discloses where said instructions comprise Reduced Instruction Set Computer (RISC) instructions (See column 22, 21-25: The processor disclosed is a RISC processor and thus will handle RISC instructions).

As per claim 41, Yates, Jr. et al discloses where said RISC instructions (See column 22, 21-25: The processor disclosed is a RISC processor and thus will handle RISC instructions) have a width of 32-bits (See column 38, lines 61-65: Yates, Jr. et al actually discloses a 64-bit processor, but it is well known in the art that backwards compatibility is built into processors and thus can handle 32-bit instructions as well), where said at least one extension has a width of 8-

bits (See column 24, lines 3-7: Variable length instructions are can be viewed as regular length instructions with extensions, include ones of 8-bits).

As per claim 42, Yates, Jr. et al discloses where said code page has a storage capacity of 4096 bytes (See column 61, lines 49-50), where said first section comprises 3072 bytes, and where said second section comprises 1024 bytes (See column 2, lines 36-49: Sections in pages are determined by flags and thus can be set to partition the 4096 bytes into 3072 and 1024 bytes).

As per claim 43, Yates, Jr. et al discloses a computer program stored on a computer readable medium (See column 19, line 65- column 20, line 1: A computer program is stored in a physical cache), said computer program comprising instructions for use with an instruction unit having a code page (See column 73, lines 28-30: Discloses the existence of code pages), comprising:

computer program code (See figure 1a and column 19, lines 41-42: A program or collection of code, is made for a computer) for partitioning said code page into at least two sections (See column 2, lines 36-49: Sections in pages are determined by flags and thus can be set to partitions and these flags must be set by some sort of instruction in the code) for storing in a first section thereof a plurality of instruction words and, in association with at least one instruction word, for storing in a second section thereof an extension to said at least one instruction word (See column 2, lines 33-36 and column 3, lines 15-20: The

instruction unit can operate with two sources of instructions and pages can be partitioned to handle two sets of instructions);

and computer program code for setting a state of at least one page table entry bit for indicating, on a code page by code page basis, whether the code page is partitioned into said first and second sections for storing instruction words and at least one instruction word extension, or whether the code page is comprised instead of a single section storing only instruction words (See column 2, lines 36-49: There are flags for both switching from one set of instructions to another and from one storage section to another and these flags must be set by some sort of instruction in the code).

As per claim 44, Yates, Jr. et al discloses further comprising computer program code for ensuring that a last instruction in said first section is a branch instruction the execution of which does not specify a target address that lies in the second section (See column 63, lines 18-21: It is recognized that page straddling branches will result in errors and thus have safeguards against the errors).

Conclusion

6. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. The following are cited to show further art with respect to a method and apparatus to extend the number of instruction bits in processors with fixed length instructions, in a manner compatible with existing code:

U.S. Patent # 5,666,510 to Mitsuishi et al shows a data processing device having expandable address space.

U.S. Patent # 5,935,237 to Chiba et al shows a microprocessor capable of carrying out different data length instructions.

U.S. Patent # 6,314,504 B1 to Dent shows a multi-mode memory addressing using variable-length.

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Vincent Lai whose telephone number is (571) 272-6749. The examiner can normally be reached on M-F 8:00-5:30 (First BiWeek Friday Off).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Fritz M. Fleming can be reached on (571) 272-4145. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Vincent Lai
Examiner
Art Unit 2181

vl
March 9, 2006

Supervisory

FRITZ FLEMING
PRIMARY EXAMINER
GROUP 2100
AU 2181

3/13/2006